

including interface for comparing the identified source elements with the corresponding translation elements, the interface receiving inputs for modifying the translation elements.

B1
5 2. (Amended) The system as recited in claim 1, wherein the source program is for a source device and the translation program is for a disparate target device.

10 3. (Amended) The system as recited in claim 1, wherein the source program is a linear assembly file for a target device and the translation program is a scheduled assembly file for that device.

Please delete Claims 4 and 5 without prejudice.

5aB Ex 15
6. (Amended) The system as recited in claim 1, wherein the translation is a context-dependent translation based on static analysis of the source program.

7. (Amended) The system as recited in claim 1, wherein the back end further comprises:

20 a translator for performing a context-dependent translation, the translator comprising:

 a translation machine description for mapping source opcodes to target opcodes;

25 a source machine description containing a description of source opcodes and source operands in a generic representation;

 a target machine description containing a description of target opcodes and target operands in a generic representation; and

30 wherein the translator receives a source instruction from said front end, utilizes the translation machine description and source machine description and target machine description to translate source elements into target elements.

8. (Amended) The system as recited in claim 7, wherein the proper target opcode is chosen from a group of potential target opcodes by comparing the target opcode and target operand with the source opcode and source operand.

SUB F1 > 9. (Amended) The system as recited in claim 7, wherein two or more source opcodes can be combined to a single target opcode when there is a target opcode that represents the two or more source code opcodes.

5
SUB D3 > 10. (Amended) The system as recited in claim 1, wherein the user interface is a graphical user interface.

10
SUB E3 > 11. (Amended) The system as recited in claim 10, wherein the graphical user interface displays at least a portion of the source elements in a source window, at least a portion of the translation elements in a translation window, and the source and translation windows are displayed side-by-side.

B2
12. (Amended) The system as recited in claim 11, wherein
15 corresponding groups of elements of the source and translation programs are aligned in the source and translation windows.

13. (Amended) The system as recited in claim 11, wherein at least one of the source and translation windows is operable to display a status icon for an
20 element in the window.

SUB D4 > 14. (Amended) A method for performing translation of an assembly language source program into an assembly language translation program, the method comprising:
25 receiving the source program;
identifying source elements in the source program;
generating the translation program having translation elements by performing a context-dependent translation of the source elements;
displaying the translation elements in an interface for receiving user inputs;
30 and
in response to user inputs, automatically regenerating selected translation elements based on the user inputs.

15. (Amended) The method as recited in claim 14, wherein the source program is for a source device and the translation program is for a disparate target device.

5 16. (Amended) The method as recited in claim 14, wherein the source program is a linear assembly program for a target device and the translation program is a scheduled assembly program for the target device.

(Please delete Claims 17 and 18 without prejudice)

10 19. (Amended) The method as recited in claim 14, further comprising: performing static analysis of the source elements in the source program; and performing context-dependent translation of the source elements based on the static analysis.

15 20. (Amended) The method as recited in claim 14, wherein the step of generating a translation program further comprises:

converting an opcode of a source machine to an opcode of a translation machine program by comparing the source opcode to possible translation opcodes;

20 converting the operand of the source opcode by comparing an operand of the source opcode in a generic expression with a generic expression for a translation operand;

combining the translation opcode and the translation operand to form a translation.

25 21. (Amended) The method as recited in claim 20, wherein the step of converting an opcode of the source file further comprises choosing a translation opcode from a group of potential translation opcodes by comparing the translation opcode and translation operand with the source opcode and source operand.

30 22. (Amended) The method as recited in claim 20, wherein the step of converting the source opcode further comprises the step of combining two or more source opcodes into a single translation opcode when there is a translation opcode that represents the two or more source opcodes.

SUB E7 23. (Amended) The method as recited in claim 14, wherein the user interface is a graphical user interface.

5 24. (Amended) The method as recited in claim 23, further comprising:
displaying the source elements in a source window;
displaying the translation elements in a translation window; and
displaying the source and translation windows side-by-side in the graphical
user interface.

B2 10 25. (Amended) The method as recited in claim 24, further comprising
aligning corresponding groups of elements of the source and translation programs in
the source and translation windows.

15 26. (Amended) The method as recited in claim 24, further comprising
displaying a status icon for an element in at least one of the source and translation
windows.

SUB D5 20 27. (Amended) A translation system for translating a source program
into a translation program, the system comprising:
a computer capable of executing a program,
an interactive program for translating code for a first processor into code for a
second processor and capable of being executed on said computer, and
a graphics interface system displaying source program elements proximate
25 corresponding translation program elements, the graphics interface system
permitting correction of the translation program elements by a user.

REMARKS

30 Claims 1-27 have been presented for examination in the above-identified U.S.
Patent Application.

Claims 1-27 have been rejected in the Office Action dated January 17, 2002.